Folding: Which to Use When

Folding

We can generalize this pattern and reduce bottom coding. We can use the empty function to be used, e.g., addition.

In the assignment, we also wrote a function to multiply up a list.

Folding

Motivation:

Examples:

Examples:

Examples:
Computation

Currying

This ability is called *currying*. Initially the number 6
\[ \text{\texttt{面貌}}: 2 \to \text{\texttt{面貌}} = \text{\texttt{面貌}} 2 \to \text{\texttt{面貌}} \]
a function that takes a number and add it to 1
\[ \text{\texttt{面貌}}: 2 \to \text{\texttt{面貌}} = \text{\texttt{面貌}} 2 \to \text{\texttt{面貌}} \]
a function that takes two numbers and add them to 1

So you can give one parameter at a time and get intermediate functions:

Currying: Examples

*Currying: Introduction*
Anonymous Functions

Motivation

Anonymous Functions: Motivation

The library has a function to do the foldl (++) True part:
foldl (++) True [map (++) True]
A function that lists all numbers in a list are negative:
map (++) True
A function that increments every number in a list:
foldl (+) 1

F. A function that squares a number is parameter:
Ex: To square a list of numbers:

Anonymous Functions

Example: Composition

Here is how a function that squares its parameter:

Anonymous Functions

Example: Composition